

CP-888 VS

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

JC720 U.S. PRO  
10/015695  
12/17/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日  
Date of Application:

2001年 7月 9日

出 願 番 号  
Application Number:

特願2001-207515

出 願 人  
Applicant(s):

沖電気工業株式会社



26694

PATENT TRADEMARK OFFICE

CERTIFIED COPY OF  
PRIORITY DOCUMENT

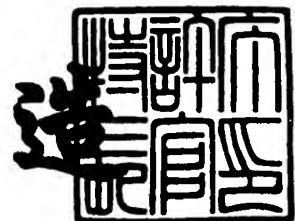
KUME  
31762-177289

12-17-01

2001年 8月31日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3078286

【書類名】 特許願

【整理番号】 CA000740

【提出日】 平成13年 7月 9日

【あて先】 特許庁長官 殿

【国際特許分類】 G06F 13/00  
G06F 13/10  
G06F 13/24

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 久米 寛司

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 本田 雄士

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 宮崎 康弘

【発明者】

【住所又は居所】 東京都港区虎ノ門1丁目7番12号 沖電気工業株式会  
社内

【氏名】 松尾 嘉勝

【特許出願人】

【識別番号】 000000295

【氏名又は名称】 沖電気工業株式会社

【代理人】

【識別番号】 100079991

【弁理士】

【氏名又は名称】 香取 孝雄

【電話番号】 03-3508-0955

【手数料の表示】

【予納台帳番号】 006895

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9001067

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 通信端末装置およびそのデータ送信方法

【特許請求の範囲】

【請求項 1】 ホスト側の制御手段に割り込みを行い、該割り込み時に該制御手段の制御に基づいて外部にデータの送信を行う通信端末装置において、該装置は、

前記データの送信において該データの連続したデータ転送を維持し、かつ該データを一時的に記憶して送出する送信バッファ手段と、

該装置における前記データの送受信状態や前記データの伝送エラーの発生に関するステータスに応じて前記ホスト側に割り込みの発生を報知する割込信号を出力する割込手段と、

前記送信バッファ手段のデータ量を監視し、該データ量の残り分を所定のデータ量として設定し、該所定のデータ量に前記データ量が達したか否かを検出し、該検出結果に応動して前記割込手段が出力する割込信号の出力制御を行うバッファ監視手段と、

前記データ量の監視に用いる情報を前記バッファ監視手段に提供し、かつ前記送信バッファ手段の動作を制御するバッファ制御手段とを含み、

前記制御手段は、前記送信バッファ手段が格納する全データ容量と前記所定のデータ量との差分で表されるデータ量を前記送信バッファ手段に転送するデータ量として送出する制御を行うことを特徴とする通信端末装置。

【請求項 2】 請求項 1 に記載の装置において、前記バッファ制御手段は、前記送信バッファ手段が送出完了状態にあっても、前記データが送出中を示している前記バッファ監視手段の動作を優先させて、前記割込手段への前記割込信号の発生制御を禁止することを特徴とする通信端末装置。

【請求項 3】 請求項 1 に記載の装置において、前記バッファ制御手段は、前記バッファ監視手段を含むことを特徴とする通信端末装置。

【請求項 4】 送信に際してデータの欠落を防止する送信バッファ手段を用意し、ホスト側に対して割り込みを行い、該割り込みに応じて前記ホスト側からの

制御により供給されるデータを前記送信バッファ手段を介してシリアル形式にして外部に送信するデータ送信方法において、該方法は、

前記送信バッファ手段に残る所定のデータ量を前記割込みを発生させるタイミング位置として設定する第 1 の工程と、

前記送信バッファ手段の全データ蓄積量と前記所定のデータ量とを考慮して前記ホスト側からデータを該送信バッファ手段に転送する第 2 の工程と、

前記送信バッファ手段に格納されたデータを読み出してシリアル変換する第 3 の工程と、

前記送信バッファ手段に残るデータ量が前記所定のデータ量になった否かの判定を行う第 4 の工程と、

前記供給されたデータがまだ前記所定のデータ量よりも多く残っているとの判定の結果に応動して前記送信バッファ手段から出力したデータ量に換算するカウント値を歩進させる第 5 の工程と、

前記送信バッファ手段に残るデータ量が前記所定のデータになったとの判定を受けて該状態に対応する前記割込みを発生させるタイミング位置にあるとし、該検出に応じて制御信号を生成する第 6 の工程と、

前記制御信号の生成を受けて割込信号を生成する第 7 の工程と、

前記ホスト側に供給された割込信号に応じてさらにデータ転送を行うか否かの判定を行う第 8 の工程と、

前記データ転送の継続を行う場合に第 2 の工程に戻り、前記データ転送を終了する場合に通信を終了させる第 9 の工程とを含むことを特徴とする通信端末装置のデータ送信方法。

【請求項 5】 請求項 4 に記載の方法において、第 6 の工程は、前記送信バッファ手段が送出完了状態にあっても、前記データの送出中が示されている場合には該データ送出中の状態を優先させて、前記割込信号の発生制御を禁止することを特徴とする通信端末装置のデータ送信方法。

【請求項 6】 請求項 4 ないし 5 のいずれか一項に記載の方法において、第 3 ないし第 7 の工程は、並行して各工程にて前記ホスト側で割込処理の有無を判定しながら他のタスク処理を行っていることを特徴とする通信端末装置のデータ

送信方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、通信端末装置およびそのデータ送信方法に関し、たとえば制御するメイン側の機器に配した非同期シリアル通信用送受信回路 (Universal Asynchronous Receiver-Transmitter: 以下、UARTという)等に適用し、このUARTによるメイン側の機器に対する割込処理と、この処理後にUARTを介してメイン側の機器とスレーブ側の機器との間で行う非同期通信の方法に用いて好適な方法に関するものである。

【0002】

【従来の技術】

UARTは、他のプロセッサの周辺装置と同様に、有益な非同期シリアル通信を行う手段として、より高速に動作でき、より集積化を図り、そして、これらを満足しながら安価に提供することを目的に進化し続けてきている。UARTは、米国IBM社のいわゆるAT (Advanced Technologies) 互換機における低速通信装置として米国ナショナルセミコンダクタ社 (National Semiconductor Corp.)の製品名8250という単独のLSI (Large-Scale Integrator)チップが採用され、拡張ボードやマザーボードに実装されたことに始まる。この後、UARTは、その性能の向上およびソフトウェアとの互換性を維持しつつ発展してきている。

【0003】

UARTは、一般にプロセッサがUARTの通信速度よりも高速にデータ通信を行えることを前提に設計されている。プロセッサがUARTにサービスを提供することは、現在作業中のプロセッサ処理を中断することになり、プロセッサにはソフトウェア上の大きな負担が強えられる。プロセッサがマルチタスクに処理を行っている場合、UARTのサービス提供を行うために割込処理を行うことになる。このような割込処理は、プロセッサの負担となってプロセッサ処理に遅延をもたらす。

【0004】

現在、これらの点を考慮し、かつ高速通信を行うために、16バイトのFIFO (Fi

rst-In First-Out)バッファを持つNS16550 (米国National Semiconductor Corp. の登録商標) が一般的に用いられている。FIFOは、送受信の双方に装備されている。

## 【 0 0 0 5 】

ところで、身近で汎用性のある無線通信手段としてブルートゥース (以下、Bluetoothという;スウェーデンのエリクソン社の登録商標) が提案された。この提案された通信規格は、1994年に自動車電話間の低コスト化および低消費電力化を図るものとして登場し、この規格はパソコンの無線カード規格等にまで拡張されてきている。この通信規格に対応したBluetooth機能を有する装置は、この装置のBluetoothにおけるホスト制御部とハードウェアモジュール部とをトランスポートバスで接続し、このバスを介してBluetoothインタフェースを実装する装置における内部の通信を行う。この通信制御は、Bluetoothのプロトコル・スタックの一つであるHCI (Host Controller Interface) レイヤで行われている。ここで、HCIは、Bluetoothにおけるデータ通信のインタフェース規格である。ハードウェアモジュール部は、配設したアンテナを通じて他のシステムにおける装置との間で送受信を行い無線データ通信を行っている。

## 【 0 0 0 6 】

ホスト制御部には、HCIドライバおよびHCIトランスポート・ドライバが実装されている。これら両ドライバは、トランスポートバスを制御する。制御内容は、主にHCIトランスポート・ドライバがハードウェアを、HCIドライバがソフトウェアを制御している。ハードウェアモジュール部は、たとえば内蔵するHCIトランスポート・ファームウェアが実装されている。HCIドライバとHCIトランスポート・ファームウェアとの間には、HCIトランスポート・ドライバを含むHCTL層 (Host Controller Transport Layer) が設けられている。HCIトランスポート・ドライバは、UARTの他に、RS (Recommended Standard)-232C、USB (Universal Serial Bus)、PCMCIA (Personal Computer Memory Card International Association)、PCI (Peripheral Component Interconnect)等が利用できるように制御する。

## 【 0 0 0 7 】

【発明が解決しようとする課題】

また、Bluetoothの通信規格において、Bluetooth Radioという無線通信では、2.4GHzのISM (Industrial Scientific Medicine)帯を使用し、拡散スペクトラム方式により通信することが規定されている。この通信におけるデータ通信の実効速度は、非対称通信時、最大で723kbpsである。非同期通信を最大のデータ通信速度で行った場合、単純に約 $1.4 \mu \text{sec/bit}$ であるから、10ビットの送信には $14 \mu \text{sec}$ を要する。

## 【 0 0 0 8 】

上述したBluetoothの構成で、たとえばUARTを介して通信する場合、プロセッサに対して割込処理を施すことにより、これにともなう遅延時間が現時点において大体 $20 \mu \text{sec}$ 生じることが知られている。したがって、割込処理を経てデータ通信を行う場合、この遅延後にデータ通信が行われることになる。

## 【 0 0 0 9 】

UARTでは、前述したように、高速通信を行うためにたとえば、NS16550（登録商標）のLSIに16バイトのFIFOを用いて通信するデータに欠落が生じないようにしている。ここで、1バイトのデータ通信には、8ビットのデータに相前後してスタートビットとストップビットが1ビットずつ配されることから、10ビットを単位に通信が行われる。これを踏まえて1バイトのFIFOによる緩衝時間を算出してみると、約 $220 \mu \text{sec}$ の緩衝時間が得られる。

## 【 0 0 1 0 】

ところで、Bluetooth機能を有する装置において割込処理を行い、UARTを介して通信を行う場合、この緩衝時間のうち、約1割が割込処理の時間に費やされてしまう。このように非同期通信が高速になればなるほどこの緩衝時間が短くなり、割込処理にともなう遅延時間は無視できないくらいに大きくなる傾向にあり、さらに将来、データ通信において問題になることが避けられない。

## 【 0 0 1 1 】

また、Bluetoothにおける非同期通信は、Bluetooth間における無線通信速度で、723bpsであり、UARTの非同期通信は、ホストのパーソナルコンピュータ等とBluetoothのホスト制御部間の有線通信規格で規定されている。たとえば、921.6kbpsをサポートしたUARTを用いて、前述した10ビットでシリアル通信を行うとき、



その最大実効通信速度は723bpsよりも低い通信速度になってしまう。Bluetooth機能を有する装置にて最大の無線通信速度を行う場合、UARTの非同期通信をより一層高速化することが望まれる。

【 0 0 1 2 】

本発明はこのような従来技術の欠点を解消し、非同期通信における有線通信の実効速度を無線通信の実効速度以上に向上させることのできる通信端末装置およびそのデータ送信方法を提供することを目的とする。

【 0 0 1 3 】

【課題を解決するための手段】

本発明は上述の課題を解決するために、ホスト側の制御手段に割り込みを行い、この割り込み時にこの制御手段の制御に基づいて外部にデータの送信を行う通信端末装置において、この装置は、データの送信においてこのデータの連続したデータ転送を維持し、かつこのデータ一時的に記憶して送出する送信バッファ手段と、この装置におけるデータの送受信状態やデータの伝送エラーの発生に関するステータスに応じてホスト側に割り込みの発生を報知する割込信号を出力する割込手段と、送信バッファ手段のデータ量を監視し、このデータ量の残り分を所定のデータ量として設定し、この所定のデータ量にデータ量が達したか否かを検出し、この検出結果に応動して割込手段が出力する割込信号の出力制御を行うバッファ監視手段と、データ量の監視に用いる情報をバッファ監視手段に提供し、かつ送信バッファ手段の動作を制御するバッファ制御手段とを含み、制御手段は、送信バッファ手段が格納する全データ容量と所定のデータ量との差分で表されるデータ量を送信バッファ手段に転送するデータ量として送出する制御を行うことを特徴とする。

【 0 0 1 4 】

本発明の通信端末装置は、バッファ監視手段でデータの送信におけるデータ量の監視に用いる情報をバッファ制御手段から受けて、設定された所定のデータ量に送信バッファ手段のデータ量が達したか否かを検出し、この検出結果に応動して割込手段に対する割込信号の出力を制御を行い、送信バッファ手段内のデータがなくなる前にホスト側に割り込みが行われることから、この割り込みから送信バ

ッファ手段への書込みまでの何も送信されない期間分を吸収してデータの書込みおよび読出しを継続させることができ、この割込みにともなって制御手段が全データ容量と所定のデータ量との差分で表されるデータ量を送信バッファ手段に供給することにより、データ転送にともなう送信バッファ手段でのデータのオーバーフローを回避している。

## 【 0 0 1 5 】

また、本発明は上述の課題を解決するために、送信に際してデータの欠落を防止する送信バッファ手段を用意し、ホスト側に対して割り込みを行い、この割込みに応じてホスト側からの制御により供給されるデータを送信バッファ手段を介してシリアル形式にして外部に送信するデータ送信方法において、この方法は、送信バッファ手段に残る所定のデータ量を割込みを発生させるタイミング位置として設定する第1の工程と、送信バッファ手段の全データ蓄積量と所定のデータ量とを考慮してホスト側からデータをこの送信バッファ手段に転送する第2の工程と、送信バッファ手段に格納されたデータを読み出してシリアル変換する第3の工程と、送信バッファ手段に残るデータ量が前記所定のデータ量になった否かの判定を行う第4の工程と、供給されたデータがまだ所定のデータ量よりも多く残っているとの判定の結果に応動して送信バッファ手段から出力したデータ量に換算するカウント値を歩進させる第5の工程と、送信バッファ手段に残るデータ量が所定のデータになったとの判定を受けてこの状態に対応する割込みを発生させるタイミング位置にあるとし、この検出に応じて制御信号を生成する第6の工程と、制御信号の生成を受けて割込信号を生成する第7の工程と、ホスト側に供給された割込信号に応じてさらにデータ転送を行うか否かの判定を行う第8の工程と、データ転送の継続を行う場合に第2の工程に戻り、データ転送を終了する場合に通信を終了させる第9の工程とを含むことを特徴とする。

## 【 0 0 1 6 】

本発明の通信端末装置のデータ送信方法は、割込みを発生させるタイミング位置として所定のデータ量を設定し、送信バッファ手段の全データ蓄積量と所定のデータ量とを考慮してホスト側からデータを送信バッファ手段に転送し、送信バッファ手段のデータを読み出し、シリアル変換して出力し、送信バッファ手段に

残るデータ量が所定のデータ量になった否かの判定し、供給されたデータがまだ所定のデータ量よりも多く残っているとの判定の結果に応動して出力済みのデータ量を表すカウント値を歩進させ、送信バッファ手段に残るデータ量が所定のデータになったとの判定を受けて割込み発生のタイミング位置にあるとして、制御信号を生成し、制御信号の生成を受けて割込信号を生成し、ホスト側に供給された割込信号に応じてさらにデータ転送を行うか否かの判定の上、データ転送の継続を行う場合にデータ転送処理に戻り、データ転送を終了する場合に通信を終了させることにより、割込みから送信バッファ手段への書込みまでの何も送信されない期間分を吸収してデータの書込みおよび読出しを継続させ、データ転送にともなうデータのオーバーフローを防止している。

【 0 0 1 7 】

## 【発明の実施の形態】

次に添付図面を参照して本発明による通信端末装置の一実施例を詳細に説明する。

【 0 0 1 8 】

本実施例は、本発明の通信端末装置を、Bluetooth機能を有する装置内のインタフェースの一つであるUART装置10に適用した場合である。本発明と直接関係のない部分について図示および説明を省略する。ここで、信号の参照符号はその現れる接続線の参照番号で表す。

【 0 0 1 9 】

図1に示すUART装置10は、あらわには示していないが、Bluetoothのホスト部12に対するBluetoothモジュール部に配設されている。UART装置10は、バスバッファ回路102、送信FIFO回路104、FIFO制御部106、シフトレジスタ108、112、受信FIFO回路110、トリガ検出部114、ステータスレジスタ116、および内部割込回路118を備えている。Bluetoothモジュール部には、図示しないがベースバンドコントローラ内にUARTの他、USB (Universal Serial Bus) や音声PCM (Pulse Code Modulation) 等やRF (Radio Frequency) モジュール、水晶発振器やフラッシュメモリ等も含まれている。また、図2に示すホスト部12には、割込回路120およびCPU (Central Processing Unit) 122が含まれている。

## 【 0 0 2 0 】

図 1 に戻って、UART 装置 10 の各部について説明する。バスバッファ回路 102 は、UART 装置 10 に配設され、バスバッファ回路 102 には、平行に装置 10 内の内部データバス 10a と装置 10 の外部のホスト部 12 に配設されたデータバス 12a とのデータ通信を確実に行わせるように入力データを一時的に格納してデータを出力するバッファメモリが配されている。バスバッファ回路 102 は、ホスト部 12 の CPU 122 から供給される各種のデータや制御データ等を内部データバス 10a を介して送信 FIFO 回路 104、FIFO 制御部 106 にそれぞれ出力する。また、バスバッファ回路 102 は、シフトレジスタ 108、受信 FIFO 回路 110、内部データバス 10a を介して供給される外部からのデータをホスト部 12 に送っている。ホスト部 12 において、この供給された受信データをデータバス 12a を介して各部に送る。

## 【 0 0 2 1 】

送信 FIFO 回路 104 には、本実施例において 16 バイトの FIFO メモリが配設されている。送信 FIFO 回路 104 は、FIFO 制御部 106 による書込み／読出し制御、パリティ情報等の管理を受けてホスト部 12 側から供給されるデータを一時的にバッファリングし、1 バイトずつ読み出したデータ 104a をシフトレジスタ 112 に出力する。送信 FIFO 回路 104 は現在の書込みポインタ (WP) 情報 104b と現在の読出しポインタ (RP) 情報 104c を、それぞれ送信制御部 106a に出力する。

## 【 0 0 2 2 】

また、受信 FIFO 回路 110 にも、本実施例では 16 バイトの FIFO メモリが配設されている。受信 FIFO 回路 110 も、FIFO 制御部 106 による書込み／読出し制御、パリティ情報等の管理を受けている。受信 FIFO 回路 110 は、シフトレジスタ 108 から供給されるデータ 108a を 1 バイトずつ取り込んで一時的にバッファリングし、受信したデータ 110a を内部データバス 10a、バスバッファ回路 102 に出力する。受信 FIFO 回路 110 は、現在の書込みポインタ情報 110b と現在の読出しポインタ情報 110c をそれぞれ、受信制御部 106b に出力する。

## 【 0 0 2 3 】

なお、送信 FIFO 回路 104 および受信 FIFO 回路 110 は、16 バイトの FIFO メモリを用いた場合を述べたが、FIFO メモリの容量はこれに限定されるものでなく、32 バイ

トや64バイトのメモリを用いてもよい。しかしながら、メモリ容量を増やして割込みにともなって生じるCPU 122の何もしない時間に対応して吸収することはできても、割込みにともなってCPU 122を占有し、その処理効率を向上させることにはつながらない。

## 【 0 0 2 4 】

FIFO制御部106には、送信制御部106aおよび受信制御部106bが含まれている。送信制御部106aは、書込みポインタ制御回路14と読出しポインタ制御回路18を含み、受信制御部106bは、書込みポインタ制御回路16と読出しポインタ制御回路20を含んでいる。書込みポインタ制御回路14と読出しポインタ制御回路18には、それぞれ送信FIFO回路104から現在の書込みポインタ104bと現在の読出しポインタ104cが供給される。同様に、受信制御部106bの書込みポインタ制御回路16と読出しポインタ制御回路20にも受信FIFO回路110から現在の書込みポインタ110bと現在の読出しポインタ110cが供給される。

## 【 0 0 2 5 】

書込みポインタ制御回路14は、次の書込みポインタの位置をどの位置にするかを示すポインタ指示信号106cとして内部データバス10a、信号線22を介して送信FIFO回路104に供給する。読出しポインタ制御回路18は、次の読出しポインタの位置をどの位置にするかを示すポインタ指示信号106cとして内部データバス10a、信号線22を介して送信FIFO回路104に供給する。受信制御部106bの書込みポインタ回路16と読出しポインタ回路20においてもそれぞれ次のポインタの位置をどの位置にするかを示すポインタ指示信号106cとして内部データバス10a、信号線110aを介して受信FIFO回路110に供給する。

## 【 0 0 2 6 】

また、書込みポインタ制御回路14は、トリガ検出部114の送信トリガ検出部114aに書込みカウント信号24を出力する。読出しポインタ制御回路18も送信トリガ検出部114aに読出しカウント信号26を出力する。受信制御部106bも受信トリガ検出部114bに書込みカウント信号28と読出しカウント信号30とをそれぞれ供給している。

## 【 0 0 2 7 】

FIFO制御部106は、このように送信FIFO回路104および受信FIFO回路110に対する書込みおよび読出しポインタを制御するだけでなく、パリティ情報等も供給され、FIFO回路104, 110の状態を調べ、この結果に応じて割込み制御も行う。FIFO制御部106は、割込み制御を行う場合などの各状態に応じたステータス信号106cを内部データバス10aを介してステータスレジスタ116に送る。

## 【 0 0 2 8 】

シフトレジスタ108は、外部から供給されるシリアルデータ32をパラレルデータ108aに変換する機能を有し、受信FIFO回路110に出力する。シフトレジスタ112は、送信FIFO回路104から供給される1バイトのデータ104aを取り込み、シリアルデータ34に変換して外部に送出する。

## 【 0 0 2 9 】

トリガ検出部114は、FIFO内のデータ残量が所定の量になったか否かを検出する機能を有している。トリガ検出部114には、送信トリガ検出部114aおよび受信トリガ検出部114bが含まれている。送信トリガ検出部114aには、図示しないがカウンタ、送信トリガレジスタ（以下、TxTRGという）14a、演算器、比較器およびトリガ出力回路（図示せず）が含まれている。TxTRG 14aは、内部レジスタである。カウンタは、書込みカウント信号24に応じてカウントし、このカウント値Nを比較器に送る。TxTRG 14aには、割込み位置であり、かつデータ残量を示す数値nを格納し、演算器に出力する。演算器は、FIFO回路として用意されたメモリ容量がFバイトのとき、送信FIFO回路104にホスト部12から転送するデータ量を算出する。この算出は、メモリ容量FとTxTRG 14aに記憶された数値nとを用い、 $(F-n)$ によって得られる。最初の割込み位置は、この供給されるデータ量に対しても作用するから、 $16-3-3+1=11$ になる。2度目以降は $2 \times 13-2=24$ 、 $3 \times 13-2=37$ 、…という位置に割込がくる。

## 【 0 0 3 0 】

したがって、割込み位置は、通し番号で表すと、 $13 \times x + 11$  ( $x=0, 1, 2, \dots$ という関係にある。比較器は、一方に供給される数値  $(F-n)$  と、他方に供給されるカウンタからの値Nとを比較する。値Nは、読出しポインタRPの位置を表している。比較器は、数値  $(F-n)$  とカウンタ値Nとが等しくなった場合に書込みポイン

タが送信トリガ位置に達したと判断して、トリガ検出信号をトリガ出力回路に出力する。比較器はこれ以外の場合には無信号を出力する。トリガ出力回路は、出力バッファとしての機能を有して、供給されたトリガ検出信号を割込出力制御信号114cを出力する。割込出力制御信号114cは、内部データバス10a、信号線32を介して内部割込回路118に供給される。

## 【 0 0 3 1 】

また、受信トリガ検出部114bにも同様に受信トリガレジスタ (RxTRG) 14bが配設され、詳細に説明しないが受信トリガ検出部114bに供給される書込みカウント信号28と読出しカウント信号30を基にトリガ検出を行い、トリガ検出信号を割込出力制御信号114cを内部割込回路118に出力する。

## 【 0 0 3 2 】

ステータスレジスタ116は、供給されるステータス信号36に割込出力制御信号38を内部割込回路118に出力する機能を有している。内部割込回路118は、割込出力制御信号114c、38が供給されると内部割込信号40をホスト部12に出力する機能を有している。

## 【 0 0 3 3 】

なお、トリガ検出部114は、FIFO制御部106と分けて説明したが、FIFO制御部106内に一体的に含めて構成することが好ましい。

## 【 0 0 3 4 】

図2の割込回路120には、内部割込信号40とホスト部12にて発生した割込要求信号42とがそれぞれ供給されている。割込回路120は、割込信号44をCPU 122に出力する。CPU 122は、供給された割込信号44に応じた制御信号46をデータバス12aを介して各部に供給する。内部割込信号40が供給された場合に、割込み処理においてCPU 122は、信号線12bを介して図1のバスバッファ回路102にデータ転送を行い、さらに、このデータがバスバッファ回路102から信号線10b、内部データバス10a、信号線22を経て送信FIFO回路104にデータ転送される。転送するデータ量は (F-n) バイトである。ここで、CPU 122には、あらかじめ割込み位置のデータである数値nが設定されている。

## 【 0 0 3 5 】

このように (F-n) バイトのデータ転送を行うことにより、送信FIFO回路104に残るデータに対して上書きされることなく、データを満杯に供給することができる。

#### 【 0 0 3 6 】

次に、UART装置10とホスト部12とのデータ転送のうち、ホスト部12からUART装置10への送信における手順を説明する。図3のフローチャートは、CPU 122の動作手順を示し、図4のフローチャートは、UART装置10における送信処理を示している。後段で説明するがCPU 122とUART装置10はそれぞれの処理を並列に行う特徴を有している。

#### 【 0 0 3 7 】

図3のフローチャートにおいて、まず、図2のCPU 122は、送信FIFO回路104に対して割込み位置を表すデータ残量として数値nをRAM (Random Access Memory ; 図示せず) に設定し、また、図1の送信トリガ検出部114aのTxTRG 14aにも数値nを設定する (ステップS10)。CPU 122では、送信FIFO回路104に転送するデータ量を考慮して転送する。転送するデータ量はデータ残量に対して空いているデータ量分である (ステップS12)。

#### 【 0 0 3 8 】

F=16バイトのFIFOメモリが配設されている場合、転送データ量は (16-n) になる。この同じ演算は送信トリガ検出部114aでも行われる。CPU 122の演算結果を送信トリガ検出部114aのTxTRG 14aに供給し、読み出して利用することができれば、送信トリガ検出部114aに演算器を設けずに済ませることができる。このとき、CPU 122および送信トリガ検出部114aは、カウント値N=0に初期設定される。この設定後、CPU 112は各タスク処理を実行する (ステップS14)。また、接続子Aを介してUART装置10の処理に進む。

#### 【 0 0 3 9 】

次に、内部割込みがあったか否かの判断をホスト部12の割込回路120で行う (ステップS16)。CPU 122に割込みがないとき (NO)、これまでの処理を継続するようにタスク処理を行う。また、内部割込みがあったとき (YES)、割込回路120は、CPU 122に割込信号44を出力する。



## 【 0 0 4 0 】

次に、転送するデータの有無に応じた判断を行う（ステップS18）。転送するデータがあるとき、データの転送を行うようにステップS12に戻って前述した処理を繰り返す。転送するデータがないとき（YES）、データの転送を終了し、データ送出を終了する。

## 【 0 0 4 1 】

一方、CPU 122の各タスクと並列にシリアル転送するUART装置10は、図4に示すように、送信FIFO回路104に格納されたデータから1バイト読み出したデータ104aをシフトレジスタ112に供給してシリアルデータに変換する。そして、シフトレジスタ112は、シリアルデータ34を外部に出力する（ステップS20）。

## 【 0 0 4 2 】

送信トリガ検出部114aでは、図示しない比較器にてカウント値Nと（F-n）値（すなわち、16-n）と比較を行う（ステップS22）。比較結果が等しくないとき（NO）、割込み位置にはまだ達していないと判断してステップS24に進む。ステップS24では、カウント値Nを+1歩進させて、以後の処理をシリアル送信に戻す。

## 【 0 0 4 3 】

また、比較結果が等しいとき（YES）、割込み位置に達したと判断して送信トリガ信号をオンにする（ステップS26）。送信トリガ検出部114aでは、送信トリガ信号のオンを受けて出力バッファ（図示せず）を介して割込出力制御信号114cを出力する（ステップS28）。なお、送信トリガ信号を割込出力制御信号114cとして出力してもよい。

## 【 0 0 4 4 】

内部データバス10aを介して供給された割込出力制御信号114cを内部割込回路118に供給し、この回路118で生成した内部割込信号40をホスト部12の割込回路120に出力する（ステップS30）。この処理の後、接続子Bを介して図3のステップS16に進む。割込回路120では、内部割込信号40の有無に応じてCPU 122に割込信号44が出力される。内部割込がある場合、転送するデータの有無を確認して割込処理を行う。

## 【 0 0 4 5 】

この動作手順についてより具体的に送信FIFO回路104の書込み／読出しの動作を基に外部へのデータ送出を説明する。送信FIFO回路104には、F=16バイトのFIFOメモリが用いられている。CPU 122や送信トリガ検出114aに割込みを発生させる位置のデータ量nを設定する。ここでデータ量n=3に設定する。送信FIFO回路104内の数字は、便宜上のデータの順番を示している。また、書込みポインタと読出しポインタはWP、RPと略しその位置を矢印で示す。

## 【 0 0 4 6 】

図5(a)の送信FIFO回路104は、初期状態にてデータが何もない状態である。WPとRPは同じ位置を示している。次に、 $(16-3)=13$ 個のデータがホスト部12から送信FIFO回路104に供給される。このデータの書込みに応じてWPは14の位置に移動する。この書込みと同時に読出しが開始される。図5(b)の送信FIFO回路104は、RP=3にある。現在の書込みポインタの位置WP=14から3つ手前の位置が送信トリガの発生位置STG (Send TriGger) である ( $14-3=11$ )。3つ手前とは、残りn=3バイトを示している。カウンタの値Nは相対的な読出しポインタの位置に対応している。

## 【 0 0 4 7 】

図5(c)では、送信FIFO回路104の読出しポインタRPが送信トリガ発生位置に達し、送信トリガ検出部114ではN=11でトリガ発生位置STGを検出する。これ以降のトリガ発生位置は後述するように13ずつずらした位置に設定される。送信トリガ検出部114でのトリガ検出を受けて内部割込回路118は、内部割込信号40を割込回路120に送出する。割込回路120は、割込信号44をCPU 122に送出する。

## 【 0 0 4 8 】

CPU 122は、転送するデータがまだ存在するか否かの判断を行い、転送データがまだある場合に、CPU 122は、書込みポインタWPから13個のデータを転送する。送信FIFO回路104は、転送の間にもデータ読出しを行って読出しポインタが13の位置にある状態を図5(d)に示している。このデータ転送後の送信FIFO回路104のトリガ発生位置STGは、書込みポインタWPの位置からデータ量3つ分戻った位置 (STG=24) になっている。

## 【 0 0 4 9 】

図5(e)の送信FIFO回路104は、読出しポインタが13回カウントされて、トリガ発生位置STGと一致した状態を示している。

【 0 0 5 0 】

ところで、FIFO制御部106は、送信FIFO回路104が送出完了状態にあっても、データが送出中を示しているトリガ検出部114の動作を優先させて、内部割込回路118への内部割込信号40の発生制御を禁止する。この禁止により、トリガ検出部114の送信トリガ検出部114aから出力される割込出力制御信号114cが有効になる。処理の上でもこの制御は有効である。

【 0 0 5 1 】

ホスト部12とUART装置10との間でデータ転送を繰り返す行う際に、図6(a)に示すように、従来のデータ送信処理は、割込み発生Iから書込みWが行われるまで、待ち時間Tが生じていた。本発明を適用したUART装置10によれば、あらかじめデータ送出が完了する前に割込み位置を規定する残りバイト数nを設定し、この設定に応じた位置に読出しポインタRPまたはカウント値Nが達したかをトリガ検出し、内部割込処理を行うように割込出力制御信号114cを出力する。このトリガ検出に応動して内部割込信号40が出力される。まだ送信FIFO回路104内のデータは送出完了にはないが、ホスト部12のCPU 122は、送信FIFO回路104内のデータに上書きをされないように、転送するデータ量を考慮して送出している。これにより、数値nの設定を理想的に設定した場合、待ち時間なく送信FIFO回路104にデータを書き込み、読み出すことができる。

【 0 0 5 2 】

しかしながら、CPU 122は、頻繁に割込処理を行うことになり、他のタスク処理との兼ね合いから処理効率が低下してしまう虞がある。CPU 122には、割込みと処理効率との間にトレードオフの関係があり、この点を考慮して残りのバイト数nを設定することが好ましい。現在、UART装置10には、64バイトまたはそれ以上の深さのFIFOメモリが用いられつつある。このような場合、本発明で指摘した残りバイト数も余裕を持って設定することができる。たとえば、数値n=8乃至16程度に設定すればよい。

【 0 0 5 3 】

また、CPU 122の高速化は、割込処理時間を少なくする要因にはなるが、CPU 122を使用するシステムが複雑化することから、CPU 122の処理時間が大きくなる。具体的な一例としては、Bluetooth等で用いられるITRON (Industrial The Real-time Operation Nucleus) のようなRTOS (Real-Time Operating System) 下でアプリケーションプログラムが実行する場合が挙げられる。このような場合、数値nは、実験的に決めることが望ましい。

## 【 0 0 5 4 】

本実施例のUART装置10は、主に、ハードウェア構成を基に説明したが、このデータ送信・設定した割込み位置の検出に応じてホスト部12からUART装置10へのデータ転送がソフトウェア的に処理できることは言うまでもない（ソフトウェアUART）。

## 【 0 0 5 5 】

以上のように構成することにより、あらかじめデータ送出が完了する前に割込み位置を設定し、この設定に応じた割込み位置に読出しポインタRPまたはカウンタ値によりが達したかをトリガ検出し、内部割込処理を行うように割込出力制御信号を出力する。このトリガ検出に応動して内部割込信号が出力される。まだ送信FIFO回路内のデータは送出完了にはないが、ホスト部のCPUは、送信FIFO回路内のデータに上書きをされないように、転送するデータ量を考慮し、CPUによる割込処理の遅延を吸収して、データを送出している。これにより、割込みにともなう無駄な時間を極力最小に抑えて従来に比べて実効効率の高い通信を行うことができる。特に、高速データ通信の要求される分野、たとえばBluetoothにおける無線通信のように高速データ通信での割込処理の遅延を相対的に小さくできる。

## 【 0 0 5 6 】

## 【発明の効果】

このように本発明の通信端末装置によれば、バッファ監視手段でデータの送信におけるデータ量の監視に用いる情報をバッファ制御手段から受けて、設定された所定のデータ量に送信バッファ手段のデータ量が達したか否かを検出し、この検出結果に応動して割込手段に対する割込信号の出力を制御を行い、送信バッ

ァ手段内のデータがなくなる前にホスト側に割り込みが行われることから、この割り込みから送信バッファ手段への書込みまでの何も送信されない期間分を吸収してデータの書込みおよび読出しを継続させることができ、この割り込みにもともなって制御手段が全データ容量と所定のデータ量との差分で表されるデータ量を送信バッファ手段に供給して、データ転送にともなう送信バッファ手段でのデータのオーバーフローを回避することにより、割り込みにもともなう無駄な時間を極力最小に抑えて従来に比べて実効効率の高い通信を行うことができる。

## 【 0 0 5 7 】

また、本発明の通信端末装置のデータ送信方法によれば、割り込みを発生させるタイミング位置として所定のデータ量を設定し、送信バッファ手段の全データ蓄積量と所定のデータ量とを考慮してホスト側からデータを送信バッファ手段に転送し、送信バッファ手段のデータを読み出し、シリアル変換して出力し、送信バッファ手段に残るデータ量が所定のデータ量になった否かの判定し、判定：否の結果に応動してカウント値を歩進させ、判定：可を受けて割り込み発生のタイミング位置にあるとして、制御信号、割込信号を順次生成し、ホスト側に供給された割込信号に応じてさらにデータ転送を行うか否かの判定の上、データ転送の継続を行う場合にデータ転送処理に戻り、データ転送を終了する場合に通信を終了させ、割り込みから送信バッファ手段への書込みまでの何も送信されない期間分を吸収してデータの書込みおよび読出しを継続させ、データ転送にともなうデータのオーバーフローを防止することにより、割り込みにもともなう無駄な時間を極力最小に抑えて従来に比べて実効効率の高い通信を行うことができる。

## 【図面の簡単な説明】

## 【図 1】

本発明の通信端末装置をUART装置に適用した概略的な構成を示すブロック図である。

## 【図 2】

図 1 の UART 装置を制御するホスト部の構成を示すブロック図である。

## 【図 3】

図 2 の CPU が行う処理手順を説明するフローチャートである。

【図 4】

図 3 の CPU と並列的に処理する UART 装置の処理手順を説明するフローチャートである。

【図 5】

図 1 の送信 FIFO 回路の動作にともなう各ポインタと割込み位置の関係を示す図である。

【図 6】

従来と本発明における割込みと書込みとの位置関係について示す図である。

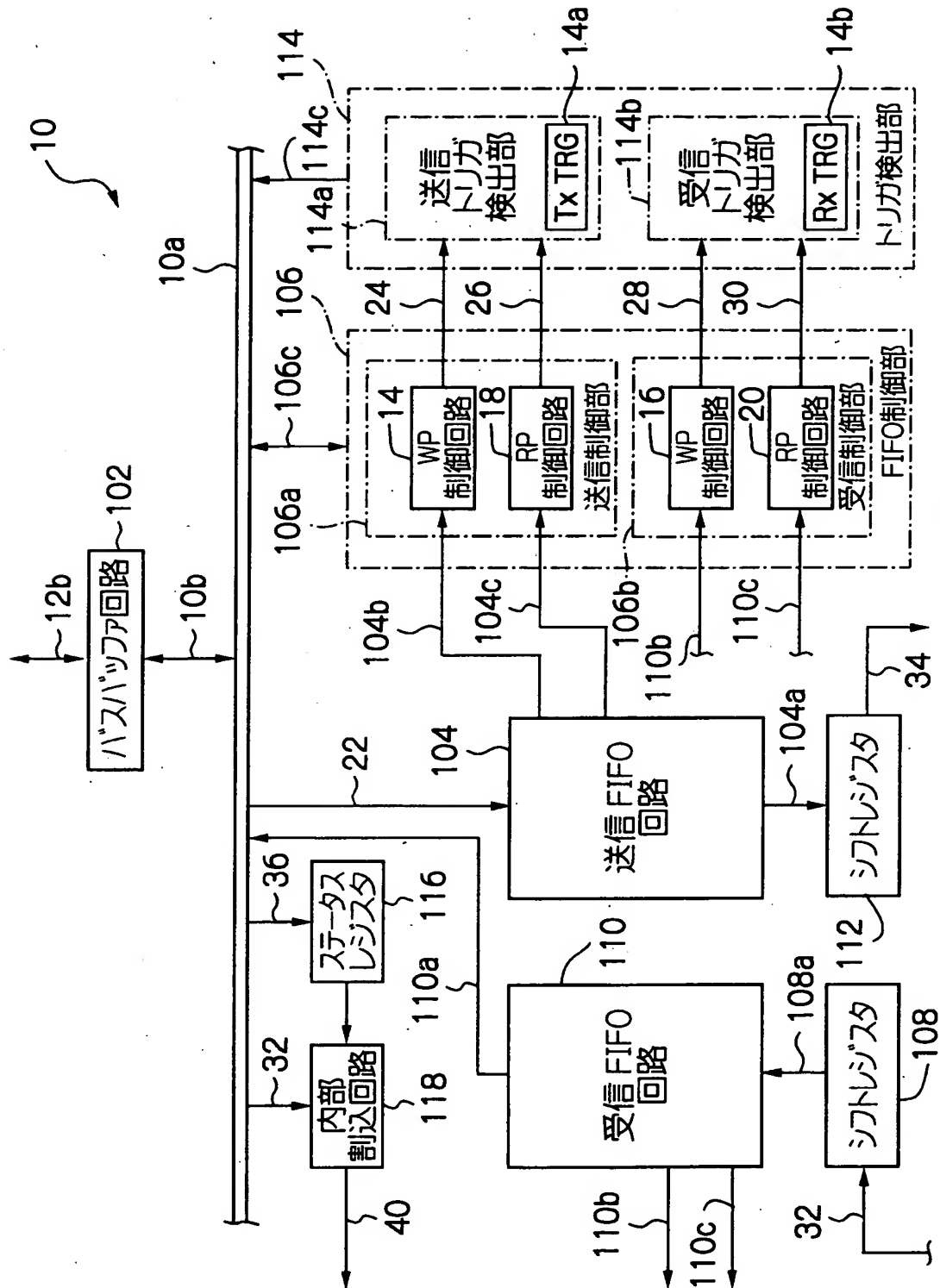
【符号の説明】

- 10 UART 装置
- 12 ホスト部
- 14a 送信トリガレジスタ (TxTRG)
- 102 バスバッファ回路
- 104 送信 FIFO 回路
- 106 FIFO 制御部
- 108, 112 シフトレジスタ
- 110 受信 FIFO 回路
- 114 トリガ検出部
- 114a 送信トリガ検出部
- 118 内部割込回路
- 120 割込回路
- 122 CPU

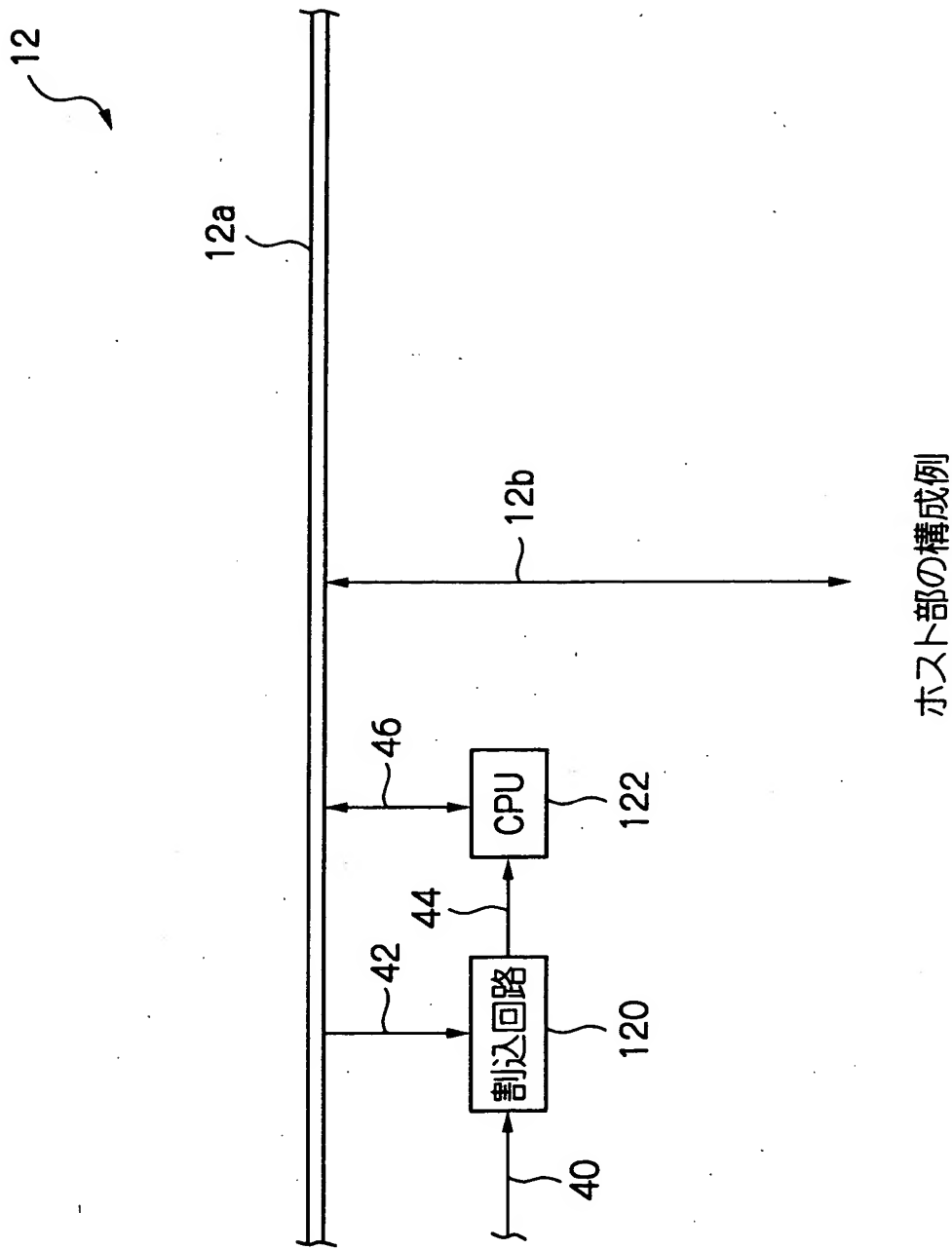
【書類名】

凶面

【図 1】

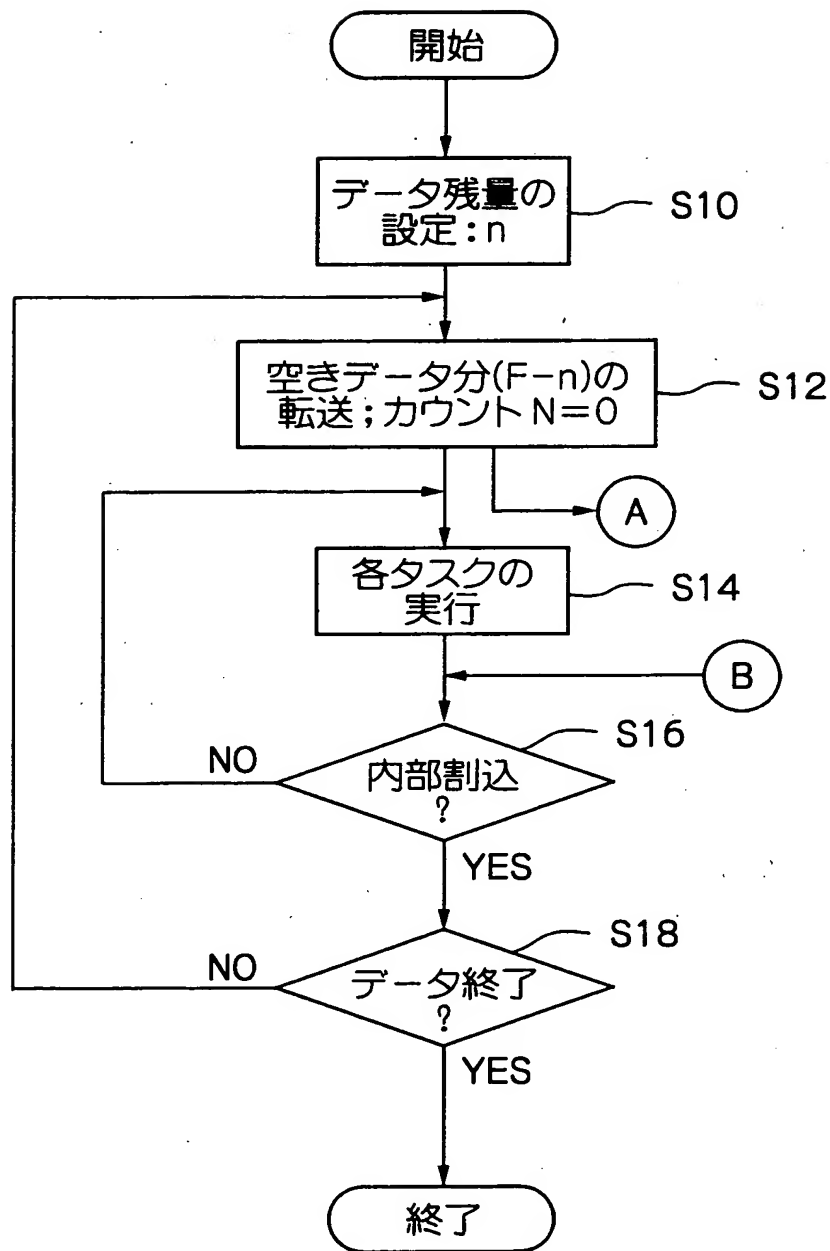


【図 2】



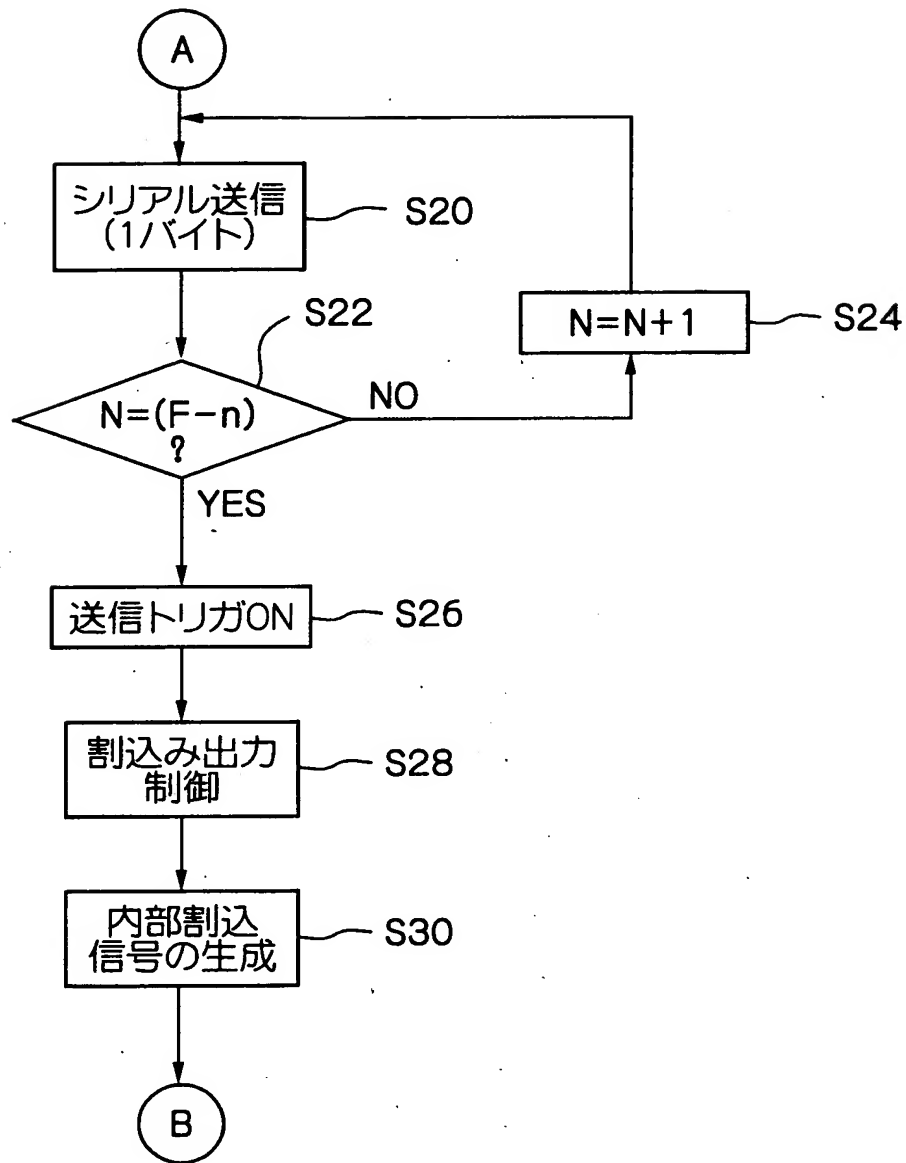


【図3】



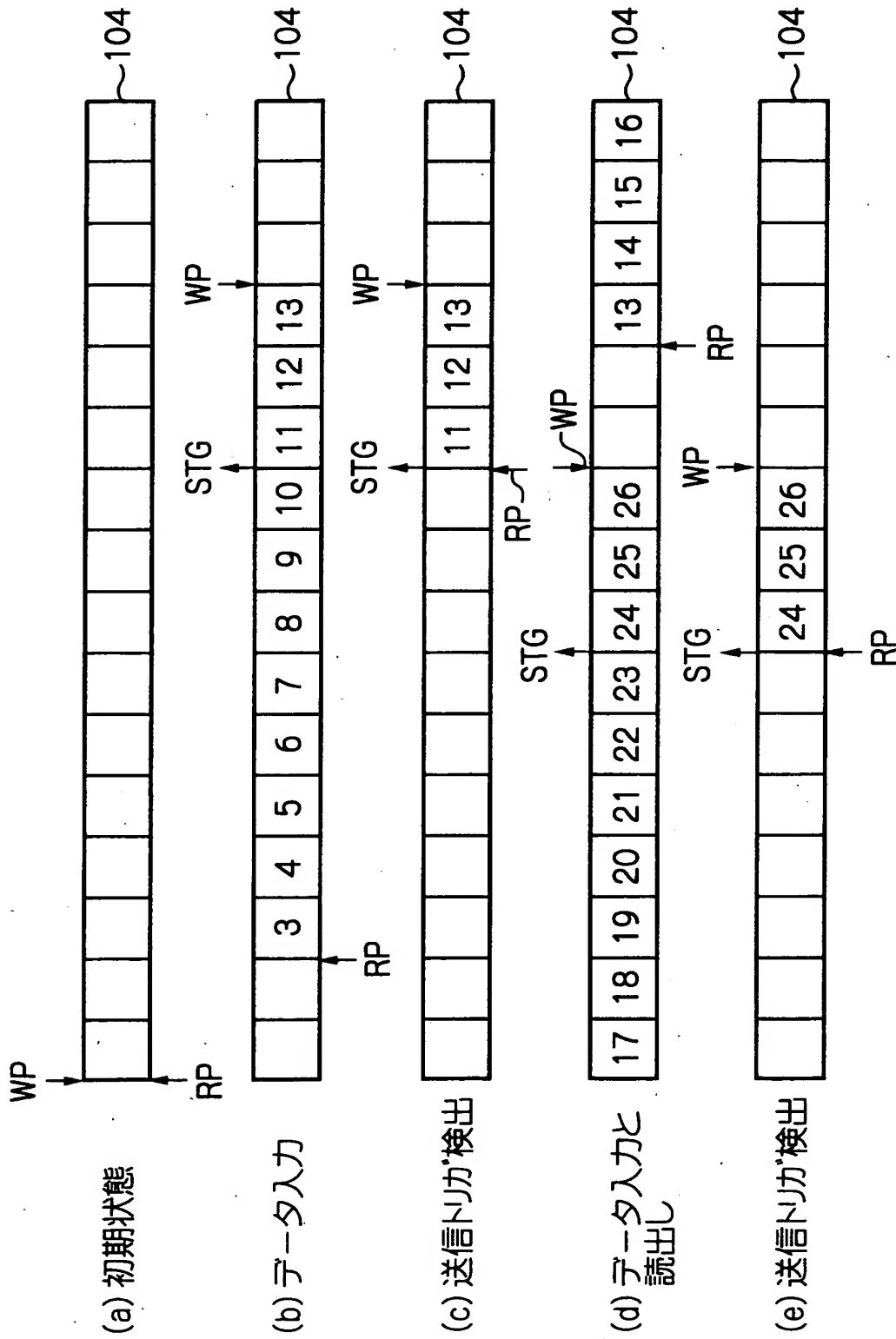
CPUの動作手順

【図 4】



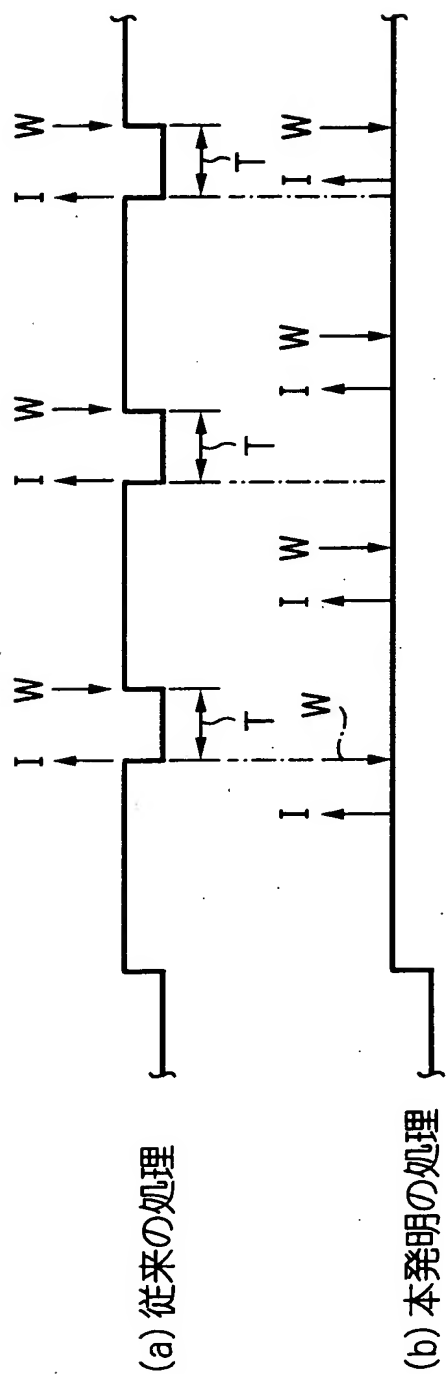
UARTの動作手順

【図 5】



送信FIFOの動作にともなう各ポインタおよび割込みの位置

【図 6】



割込みと書込みとの位置関係

【書類名】            要約書

【要約】

【課題】    非同期通信における有線通信の実効速度を無線通信の実効速度以上に向上させることのできる通信端末装置およびその通信方法の提供。

【解決手段】    UART装置10は、送信トリガ検出部114aにあらかじめデータ送出が完了する前に割込み位置を規定する残りバイト数 $n$ を設定し、この設定に応じた位置に読出しポインタ $RP$ またはカウント値 $N$ が達したかをトリガ検出し、内部割込処理を行うようにトリガ検出部114から割込出力制御信号114cを内部割込回路118に出力する。内部割込回路114は、このトリガ検出に応動して内部割込信号40を出力する。まだ送信FIFO回路104内のデータが送出完了状態にはないが、ホスト部のCPUは転送されてくるデータ量は送信FIFO回路104内のデータに上書きされないように残りバイト数 $n$ の条件を考慮して供給する。

【選択図】            図 1

出 願 人 履 歴 情 報

識別番号 [000000295]

1. 変更年月日	1990年 8月22日
[変更理由]	新規登録
住 所	東京都港区虎ノ門1丁目7番12号
氏 名	沖電気工業株式会社